I claim:

1. A method for resource management in a processor-based system, the method comprising the steps of:

storing a plurality of data values in a hardware queue, each data value being associated with a corresponding one of a plurality of resources, wherein presence of a given one of the data values in the hardware queue indicates availability of its corresponding resource to a requesting object; and

utilizing the given data value from the hardware queue to access the corresponding resource.

2. The method of claim 1, further comprising the step of reading the given data value from the hardware queue.

3. The method of claim 2, wherein the step of reading at least partially allocates the given resource by removing the given data value from the plurality of data values in the hardware queue, thereby providing access to the corresponding resource by the requesting object of a plurality of objects in the processor-based system and preventing access to the corresponding resource for other objects in the processor-based system.

4. The method of claim 2, further comprising, in conjunction with the step of reading, the step of the hardware queue removing the given data value from the plurality of data values.

5. The method of claim 1, further comprising the step of writing the given data value to the hardware queue.

6. The method of claim 5, wherein the step of writing at least partially recovers the corresponding resource by adding the given data value to the plurality of data values in the

hardware queue, thereby providing access to the given data value and its corresponding resource for any object of a plurality of objects in the processor-based system.

7. The method of claim 5, further comprising, in conjunction with the step of writing, the step of the hardware queue adding the given data value to the plurality of data values.

8. The method of claim 1, wherein each of the plurality of data values comprises a bit pattern uniquely corresponding to one of the resources.

9. The method of claim 1, wherein:

the hardware queue is accessed through a data bus;

the hardware queue comprises a queue memory and read and write interfaces;

the read interface reads from the queue memory and the write interface writes to the queue memory;

both the read and write interfaces are coupled to the data bus; and

the queue memory is configured to store the plurality of data values.

10. The method of claim 1, wherein the step of utilizing further comprises the step of mapping the given data value to its corresponding resource.

11. The method of claim 10, wherein each of the plurality of data values comprises a database key that corresponds to a portion of application data, and wherein the step of mapping further comprises the step of mapping the given data value to a corresponding portion of the application data.

12. The method of claim 11, wherein each of the database keys comprises an index, the application data comprises hardware identifiers and wherein the step of mapping further comprises the step of mapping the given data value to a corresponding hardware identifier.

13. The method of claim 11, wherein:

each of the database keys comprises an integer;

the application data comprises connection identifiers; and

the step of mapping further comprises the step of mapping the given data value to a corresponding connection identifier.

14. The method of claim 11, wherein:

the method further comprises the step of allocating a plurality of portions of a connection status table;

each of the database keys comprises an index identifying a portion of a connection status table; and

the step of mapping further comprises the step of mapping the given data value to a portion of the connection status table.

15. The method of claim 11, wherein:

each of the database keys comprises an integer corresponding to a hardware thread identification; and

the step of mapping further comprises the step of mapping the given data value to a local storage memory location corresponding to the hardware thread identification.

16. The method of claim 11, wherein:

each of the database keys comprises an index identifying an address range of a memory; and

the step of mapping further comprises the step of mapping the given data value to a given address range of the memory.

17. The method of claim 1, further comprising the steps of:

reading the hardware queue a plurality of times to retrieve a plurality of read data values from the plurality of data values;

writing the plurality of read data values to a predetermined portion of a memory; and

reading at least one of the read data values from the predetermined portion of the memory.

18. The method of claim 1, further comprising the steps of:

writing a plurality of written data values to a predetermined portion of a memory;

reading at least one of the written data values from the predetermined portion of the memory; and

writing the at least one written data value to the hardware queue to add the written data value to the plurality of data values.

19. An apparatus for resource management, the apparatus comprising:

a data bus;

a hardware queue coupled to the data bus, the hardware queue configurable to store a plurality of data values; and

one or more processors coupled to the data bus, the one or more processors adapted:

to store a plurality of data values in the hardware queue, each data value being associated with a corresponding one of a plurality of resources, wherein presence of a given one of the data values in the hardware queue indicates availability of its corresponding resource to a requesting object; and

to utilize the given data value from the hardware queue to access the corresponding resource.

20. The apparatus of claim 19, wherein the hardware queue comprises a queue memory and read and write interfaces, wherein the read interface reads from the queue memory and the write interface writes to the queue memory, wherein both the read and write interfaces are coupled to the data bus, and wherein the queue memory is configured to store the plurality of data values.

21. An integrated circuit comprising:

at least one hardware queue connectable to one or more processors via a data bus, the at least one hardware queue configurable to store a plurality of data values; and

the one or more processors adapted, for the at least one hardware queue:

to store a plurality of data values in the at least one hardware queue, each data value being associated with a corresponding one of a plurality of resources, wherein presence of a given one of the data values in the at least one hardware queue indicates availability of its corresponding resource to a requesting object; and

to utilize the given data value from the at least one hardware queue to access the corresponding resource.

22. The integrated circuit of claim 21, wherein the at least one hardware queue comprises a plurality of hardware queues, each of the plurality of hardware queues being associated with a given set of resources of a plurality of sets of resources.

23. The integrated circuit of claim 21, wherein the one or more processors are adapted to read the given data value from the at least one hardware queue during a read operation, wherein the hardware queue is adapted, in conjunction with the read operation, to remove the given data value from the plurality of data values, thereby providing access to the corresponding resource by the requesting object of a plurality of objects and preventing access to the corresponding resource for other objects.

24. The integrated circuit of claim 21, wherein the one or more processors are adapted to write the given data value to the at least one hardware queue during a read operation, wherein the at least one hardware queue is adapted, in conjunction with the write operation, to add the given data value to the plurality of data values, thereby providing access to the corresponding resource by providing access to the given data value.

25. The integrated circuit of claim 21, wherein the one or more processors are further adapted, in conjunction with utilizing the given data value, to map the given data value to its corresponding resource.

26. An integrated circuit comprising:

a data bus;

at least one processor coupled to the data bus;

at least one hardware queue coupled to the data bus, the at least one hardware queue comprising:

a queue memory configurable to store a plurality of data values;

a read interface coupled to the data bus and adapted to read from the queue memory; and

a write interface coupled to the data bus and adapted to write to the queue memory.

27. The integrated circuit of claim 26, wherein the at least one hardware queue is adapted to remove, in conjunction with a read operation, a given data value from the plurality of data values so that the given data value can no longer be read from the plurality of data values, and wherein the at least one hardware queue is adapted to add, in conjunction with a write operation, a given data value to the plurality of data values so that the given data value is accessible from the plurality of data values.

28. The integrated circuit of claim 26, wherein the at least one processor is adapted:

to store the plurality of data values in the at least one hardware queue, each data value being associated with a corresponding one of a plurality of resources, wherein presence of a given one of the data values in the at least one hardware queue indicates availability of its corresponding resource to a requesting object; and

to utilize the given data value from the at least one hardware queue to access the corresponding resource.

29. The integrated circuit of claim 28, wherein the integrated circuit comprises the plurality of resources, the plurality of resources are external to the integrated circuit, or at least part of each resource of the plurality of resources resides on the integrated circuit.

30. The integrated circuit of claim 26, further comprising:

an address bus; and

at least one queue enable module coupled to the address bus and to the at least one hardware queue, the at least one queue enable module adapted to enable the at least one hardware queue when an address on the address bus corresponds to an address associated with the at least one hardware queue.